Appl. No. 10/774,267

In the Claims

1-3. (Cancelled)

4. (Original) A method of transferring a transaction processing workload, the method comprising:

establishing a plurality of primary process pairs on a first node in a cluster of nodes, each node including one or more servers and being associated with a plurality of stable storage volumes, the plurality of primary process pairs including a plurality of first primary processes and a plurality of first backup processes;

grouping the plurality of stable storage volumes of the first node into a number of groups and assigning a separate audit trail to each group of stable storage volumes, each of the separate audit trails and their associated group of stable storage volumes forming a log storage group, the separate audit trail for each log storage group recording stable storage updates for the stable storage volumes in the log storage group;

establishing a plurality of backup process pairs on a second node in the cluster, the plurality of backup process pairs including a plurality of second primary processes and a plurality of second backup processes;

if the node on which the plurality of backup process pairs is established is in communication with the node hosting the plurality of primary process pairs, performing checkpointing operations via the network from the plurality of primary process pairs to the plurality of backup process pairs;

performing checkpointing operations on the node hosting the plurality of primary process pairs;

detecting a failure that makes the first node inoperable

or inaccessible;

after detecting the failure, engaging the plurality of backup process pairs as a new plurality of primary process pairs to perform the transaction processing workload of the plurality of primary process pairs in the failed node, the plurality of backup process pairs being configured to operate with a log storage group associated with the plurality of primary process pairs on the failed node.

5. (Original) A method of transferring a transaction processing workload as recited in claim 4, wherein prior to engaging the backup process pair, if the plurality of backup process pairs is not present, the method further comprises creating a plurality of backup process pairs on the second node.

6. (Original) A method of transferring a transaction processing workload as recited in claim 4,

wherein the node that hosts the plurality of new primary process pairs has a cache for holding transaction data; and

wherein the cache is loaded with transaction data derived from the audit trail to prepare the cache for operation with the plurality of new primary process pairs.

7. (Original) A method of transferring a transaction processing workload as recited in claim 4,

wherein performing checkpointing operations on the node hosting the plurality of primary process pairs includes:

checkpointing transaction updates from the plurality of first primary processes to the plurality of first backup processes;

writing transaction updates and a communications flag to the audit trail, the communications flag indicating whether

the node hosting the plurality of primary process pairs is in communication with the node on which the plurality of backup process pairs is established; and

writing transaction updates to said plurality of stable storage volumes.

8. (Original) A method of transferring a transaction processing workload as recited in claim 4, wherein performing checkpointing operations from the plurality of primary process pairs to the plurality of backup process pairs includes:

checkpointing transaction updates from the plurality of primary process pairs to the plurality of backup process pairs;

writing transaction updates and a communications flag to the audit trail, the communications flag indicating whether the node hosting the plurality of primary process pairs is in communication with the node on which the plurality of backup process pairs is established; and

writing transaction updates to said plurality of stable storage volumes.

9. (Original) A method of transferring a transaction processing workload as recited in claim 4,

wherein the node hosting the plurality of primary process pairs includes a primary audit process pair that performs logging operations for the plurality of primary process pairs;

wherein the node on which the plurality of backup process pairs is established has a backup audit process pair that performs logging operations for the plurality of backup process pairs after takeover; and

further comprising checkpointing audit trail information from the primary audit process pair to the backup process pair.

10. (Original) A method of transferring a transaction processing workload as recited in claim 4,

wherein the node hosting the plurality of primary process pairs includes a primary audit process pair that maintains a list of sequential updates in an audit trail for the plurality of primary process pairs;

wherein the node on which the plurality of backup process pairs is established has a backup audit process pair that maintains a list of sequential updates in an audit trail for the plurality of backup process pairs;

further comprising writing each update in the audit trail to a volume in the log storage group if the node on which the plurality of backup process pairs is established is out of communication with the node hosting the plurality of primary process pairs.

11. (Original) A method of transferring a transaction processing workload as recited in claim 4,

wherein the node that hosts the plurality of new primary process pairs has a cache for holding transaction data;

wherein, if, based on the communication flag in the audit trail, the node on which the plurality of backup process pairs is established was out of communication with the node hosting the plurality of primary process pairs prior to the failure, said engaging the plurality of backup process pairs includes performing a recovery process based on data in the audit trail of the log storage group to prepare the cache for operation with the plurality of new primary process pairs.

12. (Original) A method of transferring a transaction processing workload as recited in claim 11,

wherein the node that hosts the plurality of new primary process pairs has a cache for holding transaction data; and

wherein performing a recovery procedure includes performing redo and undo operations based on updates in the audit trail of the log storage group to prepare the cache.

13. (Original) A method of transferring a transaction processing workload as recited in claim 12,

wherein the node that hosts the plurality of new primary process pairs has a cache for holding transaction data and the cache is operated according to a "steal, no force" policy; and

wherein the audit trail includes at least two checkpoints and the undo and redo operations, performed based on the updates in the audit trail, stop at the penultimate checkpoint.

14. (Original) A method of transferring a transaction processing workload as recited in claim 4,

wherein the node that hosts the plurality of new primary process pairs has a cache for holding transaction data; and

wherein, if the plurality of backup process pairs is not present, said engaging the plurality of backup process pairs includes performing a recovery process based on data in the audit trail of the log storage group to prepare the cache for operation with the plurality of new primary process pairs.

15. (Original) A method of transferring a transaction processing workload as recited in claim 14, wherein performing a recovery procedure includes performing redo and undo operations based on updates in the audit trail of the log storage group.

16. (Original) A method of transferring a transaction processing workload as recited in claim 15,
wherein the node that hosts the plurality of new primary

process pairs has a cache for holding transaction data and the cache is operated according to a "steal, no force" policy; and wherein the redo and undo operations, performed based on the updates in the audit trail in the log storage group, stop at the beginning of the audit trail.

17. (Currently amended) A transaction processing apparatus, comprising:

a plurality of stable database storage volumes being organized into a plurality of stable storage volume groups ~~, one or more of the storage volumes being organized into a group;~~

a plurality of stable transaction log storage volumes, each log storage volume being associated with one of said stable storage volume groups to form a log storage group, each log storage volume having a separate audit trail to record stable storage updates for stable storage volumes in an associated stable storage volume group; and

a plurality of connected computing nodes, at least one node having a plurality of primary process pairs for performing work on behalf of a transaction by accessing the log storage group, wherein any of the stable storage volumes and log storage volumes are accessible by any of said computing nodes, and at least one other node having a plurality of backup process pairs for taking over the work of said plurality of primary process pairs by accessing the log storage group used by said plurality of primary process pairs, if said plurality of primary process pairs are non-operational.

18. (Original) A transaction processing apparatus as recited in claim 17, wherein the plurality of primary process pairs communicate checkpointing information to the plurality of

backup process pairs.

19. (Original) A transaction processing apparatus as recited in claim 17,

wherein the computing node with the plurality of backup process pairs has a cache for holding transaction data; and

wherein the cache is loaded with transaction data derived from the log storage volume of the log storage group to prepare the cache for use by the plurality of backup process pairs when the plurality of backup pairs takes over the work of the plurality of primary process pairs.

20. (Original) A transaction processing apparatus as recited in claim 17,

wherein the computing node with the plurality of primary process pairs and the computing node with the plurality of backup process pairs each have a cache for holding transaction data, the cache in the computing node with the plurality of backup process pairs being maintained with substantially the same information as the cache in the computing node with the plurality of primary process pairs; and

wherein the cache in the computing node of the plurality of backup process pairs is used by the plurality of backup process pairs when the plurality of backup pairs takes over the work of the plurality of primary process pairs.

21-22. (Cancelled)